

THREE.JS

Einführung & Beispiele für three.js | Sören Knöll | Dataviz Challenges April 2013

RENDERER:

HTML 5 canvas, WebGL oder SVG

DAS BESTE ERGEBNIS MIT:

1. Chrome

2. Firefox

BASIC STEPS:

1. scene

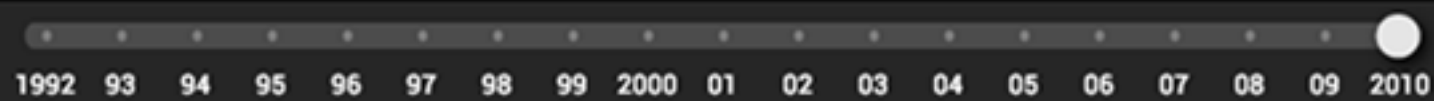
2. renderer

3. camera

4. objects | materials







1 H Hydrogen 1.00794																	2 He Helium 4.002602			
3 Li Lithium 6.941	4 Be Beryllium 9.012182											5 B Boron 10.811	6 C Carbon 12.0107	7 N Nitrogen 14.0067	8 O Oxygen 15.9994	9 F Fluorine 18.9984032	10 Ne Neon 20.1797			
11 Na Sodium 22.98976928	12 Mg Magnesium 24.304											13 Al Aluminium 26.9815386	14 Si Silicon 28.0855	15 P Phosphorus 30.973762	16 S Sulfur 32.065	17 Cl Chlorine 35.453	18 Ar Argon 39.948			
19 K Potassium 39.0983	20 Ca Calcium 40.078	21 Sc Scandium 44.955912	22 Ti Titanium 47.867	23 V Vanadium 50.9415	24 Cr Chromium 51.9961	25 Mn Manganese 54.938045	26 Fe Iron 55.845	27 Co Cobalt 58.933196	28 Ni Nickel 58.6934	29 Cu Copper 63.546	30 Zn Zinc 65.38	31 Ga Gallium 69.723	32 Ge Germanium 72.64	33 As Arsenic 74.9216	34 Se Selenium 78.96	35 Br Bromine 79.904	36 Kr Krypton 83.796			
37 Rb Rubidium 85.4678	38 Sr Strontium 87.62	39 Y Yttrium 88.90585	40 Zr Zirconium 91.224	41 Nb Niobium 92.90638	42 Mo Molybdenum 95.94	43 Tc Technetium (98)	44 Ru Ruthenium 101.07	45 Rh Rhodium 102.9055	46 Pd Palladium 106.42	47 Ag Silver 107.8682	48 Cd Cadmium 112.411	49 In Indium 114.818	50 Sn Tin 118.71	51 Sb Antimony 121.76	52 Te Tellurium 127.6	53 I Iodine 126.90447	54 Xe Xenon 131.293			
55 Cs Cesium 132.9054	56 Ba Barium 137.327			72 Hf Hafnium 178.49	73 Ta Tantalum 180.94788	74 W Tungsten 183.84	75 Re Rhenium 186.207	76 Os Osmium 190.23	77 Ir Iridium 192.222	78 Pt Platinum 195.084	79 Au Gold 196.966569	80 Hg Mercury 200.59	81 Tl Thallium 204.3833	82 Pb Lead 207.2	83 Bi Bismuth 208.9804	84 Po Polonium (209)	85 At Astatine (210)	86 Rn Radon (222)		
87 Fr Francium (223)	88 Ra Radium (226)			104 Rf Rutherfordium (261)	105 Db Dubnium (268)	106 Sg Seaborgium (271)	107 Bh Bohrium (272)	108 Hs Hassium (277)	109 Mt Meitnerium (276)	110 Ds Darmstadtium (281)	111 Rg Roentgenium (282)	112 Cn Copernicium (285)	113 Uut Ununtrium (284)	114 Fl Flerovium (289)	115 Uup Ununpentium (288)	116 Lv Livermorium (293)	117 Uus Ununseptium (294)	118 Uuo Ununoctium (294)		
		57 La Lanthanum 138.90547	58 Ce Cerium 140.12	59 Pr Praseodymium 140.90768	60 Nd Neodymium 144.242	61 Pm Promethium (145)	62 Sm Samarium 150.36	63 Eu Europium 151.964	64 Gd Gadolinium 157.25	65 Tb Terbium 158.92535	66 Dy Dysprosium 162.5	67 Ho Holmium 164.93032	68 Er Erbium 167.259	69 Tm Thulium 168.93421	70 Yb Ytterbium 173.054	71 Lu Lutetium 174.967				
		89 Ac Actinium (227)	90 Th Thorium 232.0377	91 Pa Protactinium 231.036	92 U Uranium 238.02891	93 Np Neptunium (237)	94 Pu Plutonium (244)	95 Am Americium (243)	96 Cm Curium (247)	97 Bk Berkelium (247)	98 Cf Californium (251)	99 Es Einsteinium (252)	100 Fm Fermium (257)	101 Md Mendelevium (258)	102 No Nobelium (259)	103 Lr Lawrencium (260)				

TABLE

SPHERE

HELIX

GRID



RENDERER:

HTML 5 canvas, WebGL oder SVG

DAS BESTE ERGEBNIS MIT:

1. Chrome

2. Firefox

BASIC STEPS:

1. scene

2. renderer

3. camera

4. objects | materials


```
// set the scene size
var WIDTH = 400,
    HEIGHT = 300;

// set some camera attributes
var VIEW_ANGLE = 45,
    ASPECT = WIDTH / HEIGHT,
    NEAR = 0.1,
    FAR = 10000;

// get the DOM element to attach to
// - assume we've got jQuery to hand
var $container = $('#container');

// create a WebGL renderer, camera
// and a scene
var renderer = new THREE.WebGLRenderer();
var camera = new THREE.PerspectiveCamera( VIEW_ANGLE,
                                          ASPECT,
                                          NEAR,
                                          FAR );
var scene = new THREE.Scene();

// the camera starts at 0,0,0 so pull it back
camera.position.z = 300;

// start the renderer
renderer.setSize(WIDTH, HEIGHT);

// attach the render-supplied DOM element
$container.append(renderer.domElement);

// create the sphere's material
var sphereMaterial = new THREE.MeshLambertMaterial(
{
    color: 0xCC0000
});

// set up the sphere vars
var radius = 50, segments = 16, rings = 16;

// create a new mesh with sphere geometry -
// we will cover the sphereMaterial next!
var sphere = new THREE.Mesh(
    new THREE.SphereGeometry(radius, segments, rings),
    sphereMaterial);

// add the sphere to the scene
scene.add(sphere);

// and the camera
scene.add(camera);

// create a point light
var pointLight = new THREE.PointLight( 0xFFFFFF );

// set its position
pointLight.position.x = 10;
pointLight.position.y = 50;
pointLight.position.z = 130;

// add to the scene
scene.add(pointLight);

// draw!
renderer.render(scene, camera);
```

INTERESSANT & HILFREICH

www.mrdoob.com

www.alteredqualia.com

EXPLIZIT

<http://mrdoob.github.com/three.js>

www.github.com/mrdoob/three.js/issues

www.aerotwist.com/tutorials/getting-started-with-three-js